



PATENT
1509-453

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant: Thomas FRIETSCH : Confirmation No. 7146
Serial No.: 10/698,016 : Art Unit: 2153
Filed: October 31, 2003 : Examiner: Not assigned
For: NETWORK SERVER AND METHOD OF DISCOVERY OF A
NETWORK NODE

SUBMISSION OF PRIORITY DOCUMENT

Commissioner for Patents
P. O. Box 1450
Alexandria, VA 22313-1450

April 2, 2004

Sir:

At the time the above-identified application was filed, Applicant claimed priority in accordance with the provisions of 35 U.S.C. §119 to German Application No. 103 38 113.9, filed August 15, 2003. A copy of the priority application is attached.

If necessary, the Commissioner is hereby authorized to charge any omitted fees due in connection with the filing of this paper, including extension of time fees, to counsel's Deposit Account 07-1337.

Respectfully submitted,

LOWE HAUPTMAN GILMAN & BERNER, LLP

By: 

Allan M. Lowe, Reg. No. 19,641

1700 Diagonal Road, Suite 310
Alexandria, VA 22314
Telephone: (703) 684-1111
Facsimile: (703) 518-5499
AML:rk

BUNDESREPUBLIK DEUTSCHLAND



Prioritätsbescheinigung über die Einreichung einer Patentanmeldung

Aktenzeichen: 103 38 113.9

Anmeldetag: 15. August 2003

Anmelder/Inhaber: Hewlett-Packard Company, A Delaware Corporation,
Palo Alto, Calif./US

Bezeichnung: Network server and method of discovery of a network
node

IPC: H 04 L 12/24

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ur-
tümlichen Unterlagen dieser Patentanmeldung.

München, den 28. Oktober 2003
Deutsches Patent- und Markenamt
Der Präsident
Im Auftrag

Schäfer

Quermann & Richardt • Unter den Eichen 7 • D-65195 Wiesbaden

Deutsches Patent- und
Markenamt
Postfach

80297 München

Patentanwälte

European Patent Attorneys
European Trademark Attorneys

Diplom-Ingenieur
Helmut Quermann

Diplom-Ingenieur
Markus Richardt

August 15, 2003
MR/DV/B

Attorney docket: hewl.203.04 DE

Hewlett-Packard Company, A Delaware Corporation
3000 Hanover Street Palo Alto, California 94304
USA

Network server and method of discovery of a network node

Description

Field of the invention

- 5 This invention relates to computer systems and more particularly to computer networks that interconnect network nodes, such as computers and computer peripherals. Even more particularly, the invention relates to discovery of network nodes.

Background and prior art

5 Network Management Systems like the OpenView Network Node Manager product are designed to discover network topology (i.e., a list of all network node in a domain, their type, and their connections), monitor the health of each network node, and report problems to the network administrator. OpenView Network Node Manager (NNM) is a product distributed by Hewlett-Packard Company of Palo Alto, Calif.

10

The monitoring function of such a system is usually performed by a specialized computer program which periodically polls each network element and gathers data which is indicative of the network element's health. A monitor program typically runs on a single host. However, in distributed networks, monitors may
15 run on various nodes in the network, with each monitor reporting its results to a centralized display.

20

Network discovery is performed periodically using internet control message protocol (ICMP) polling (pings and mask requests), simple network management protocol (SNMP) polling and / or other diagnostic requests. In addition or as an alternative to these protocols Desktop Management Interface (DMI) or Windows
Management Instrumentation (WMI) which is also referred to as WBEM can be used.

25

For example by means of SNMP polling a configuration check operation is performed on each node. By default this configuration check operation is performed once each day whereby this period is configurable. This way it is determined at periodic intervals if any relevant configuration information has changed since the last check. More technical detail of the internals of the NNM product is
30 disclosed on (http://support.openview.hp.com/pdf/dev_ov_netmon96.pdf), the entirety of which is herein incorporated by reference.

Elements of the code for discovering the topology of a plurality of network elements and the code for periodically polling a plurality of network interfaces associated with the plurality of network elements are also disclosed in U.S. Pat. No. 5,185,860 of Wu entitled "Automatic Discovery of Network Elements, and in
5 U.S. Pat. No. 5,276,789 of Besaw et al. entitled "Graphic Display of Network Topology". Both of these patents are hereby incorporated by reference for all that they disclose.

10 A disadvantage of scheduled discovery which is performed by the NNM product is that the discovery procedure can negatively affect network performance when the discovery procedure is carried out. This is why the discovery schedule is typically set such that the discovery procedure is carried out when the network load is low, i.e. during the night. However, performing the discovery at pre-
15 determined time intervals has the disadvantage that network nodes which are connected to the network only temporarily can be missed by the discovery. In particular this applies to portable computers, such as lap top computers which are frequently connected and disconnected to the network by means of a docking station.

20 **Summary of the invention**

The present invention provides for a method of discovery of a network node. Initially an access request is received from a network node. In response to the access request a discovery request is generated and transmitted to a discovery
25 server. The discovery server performs a discovery procedure for the network node from which the access request has been received.

In comparison to a scheduled discovery this has the advantage that network nodes, which are only temporarily connected to the network, are always discovered while making minimal usage of network bandwidth resources for the discovery procedure as not the whole network but only the network node identified
30 by the access request is discovered. Another advantage is that no extra soft-

ware needs to be installed on the network nodes as the discovery procedure can be initiated by a standard access request, such as a log on request or an internet protocol (IP) address request.

- 5 A further advantage is that the unplanned discovery of network nodes can be performed centrally by one discovery server maintaining the discovery rules and collecting the discovery results or a few interconnected discovery servers. This greatly simplifies the network administration task.

- 10 In accordance with a preferred embodiment of the invention a discovery procedure is only performed for a given network node after a minimum amount of time has lapsed since the last discovery procedure had been performed. This way unnecessary discovery procedures are avoided when a network node is frequently disconnected and reconnected to the computer network. For example, the configuration of a lap top computer usually does not change very often. However, a lap top computer system can be removed and reconnected to the computer network within short time intervals, especially if a docking station is used.

- 20 In accordance with a further preferred embodiment of the invention the discovery procedure is initiated by a log on request received by a network server, such as a domain controller, from a network node. Typically the user of the network node needs to enter his or her user ID and password for authentication. After the log on procedure has been completed successfully a discovery request is generated and sent to a discovery server from the network server.

- In accordance with a further preferred embodiment of the invention the network node request which initiates the discovery procedure is an IP address request. For example, when a computer peripheral, such as a printer, is connected to the computer network, the computer peripheral sends an IP address request which is processed by a DHCP server. The DHCP server assigns an IP address to the requesting network node. After the IP address has been assigned to the net-

work node a discovery request is generated and transmitted to the discovery server together with the IP address.

Brief description of the drawings

5

In the following preferred embodiments of the invention will be described, by way of example, and with reference to the drawings in which:

Figure 1 is illustrative of a block diagram of a computer network having a network server being coupled to a discovery server,

10

Figure 2 is illustrative of a flow chart for performing a method of discovery of a network node,

15 Figure 3 is a block diagram of a computer network having a temporarily connected portable computer,

Figure 4 is illustrative of a block diagram of a computer network having a DHCP server,

20

Figure 5 is illustrative of a flow chart of an alternative method of discovery of the network node.

Detailed description

25

Figure 1 shows a network server 100 of a computer network having a plurality of network nodes. For ease of explanation only network node 102 of the computer network is shown in figure 1. Network node 102 is not permanently coupled to the computer network but is disconnected and reconnected from time to time. Network node 102 has an assigned IP address for addressing of network node 102 by means of the transmission control protocol / internet protocol (TCP / IP).

30

Network server 100 has program component 104 for processing of access request 106 received from network node 102. For example program component 104 serves for authentication purposes of network node 102 and / or of a user of network node 102.

Further network server 100 has program component 108 which serves for initiation of a discovery procedure. After successful completion of the processing of access request 106 by program component 104, program component 108 generates discovery request 110 and sends discovery request 110 together with the IP address of network node 102 to discovery server 112.

Discovery server 112 has memory 114 for storing of the IP address received with discovery request 110. For example memory 114 is a FIFO buffer for storing a stack of IP addresses.

Discovery server 112 has discovery program 116 for performing a discovery procedure for network node 102. The discovery procedure is performed by ICMP and / or SNMP polls 118 in order to determine the network topology, i.e. other nodes to which network node 102 is connected, to discover configuration information of network node 102 and / or other status information of network node 102.

In operation, when network node 102 is physically connected to the network, it sends access request 106 to network server 100 in order to obtain access to the network resources. When access request 106 is received by network server 100 this invokes program component 104. Program component 104 processes access request 106. For example, program component 104 performs an authentication procedure prior to granting access to the network resources. After successful completion of the processing of access request 106 by program component 104 access is granted to network node 102 and program component 108 is invoked by program component 104.

Program component 108 generates discovery request 110 which comprises the IP address of network node 102[brujon1]. It is to be noted that in the embodiment considered here fixed IP addresses are being used, i.e. the IP address of network node 102 is assigned once and then fixed.

Program component 108 sends discovery request 110 with the IP address to discovery server 112. Discovery server 112 stores the IP address received with discovery request 110 in memory 114. Further discovery program 116 is invoked for performing a discovery procedure for network node 102 which is identified by its IP address. This is performed by ICMP and / or SNMP polls.

Discovery server 112 can receive a sequence of discovery requests 110 containing various IP addresses of various network nodes which have requested access to the computer network. This sequence of IP addresses is stored as a stack in memory 114. The stack of IP addresses is processed by discovery program 116 in first-in-first-out order.

It is to be noted that network server 100 and discovery server 112 are implemented on different server computers; alternatively network server 100 and discovery server 112 are implemented on the same server computer which is partitioned correspondingly. In particular the discovery server 112 can be implemented by any server computer running the relevant program, such as the network server computer 100.

25

Figure 2 shows a corresponding flowchart. In step 200 an access request is received from a removable network node which has been physically reconnected to the network. The access request from the network node is directed towards logical re-coupling of the network node to the computer network, i.e. the granting of access to network resources.

In step 202 the access request is processed. For example prior to granting of access the authenticity of the network node and / or of its user and / or of its

access rights are checked. After successful authentication and / or successful check of the access rights the network node is granted access to the network and is logically reconnected.

5 In response to the logical reconnection of the network node a discovery request is generated and sent to the discovery server. The discovery request contains an identifier of the network node, such as its node ID. Preferably the IP address of the network node is used as a node ID.

10 Alternatively the name or other addressing information of the node can be used as an identifier. In response to the discovery request the discovery server performs a discovery procedure for the network node identified by the node ID in the discovery request in step 206.

15 The trigger for performing the discovery procedure is the discovery request and not a prescheduled point of time. This means that a discovery procedure is only initiated if the network node is actually coupled to the network.

Figure 3 is an alternative embodiment based on the embodiment of figure 1.

20 Like elements of the embodiments of figures 1 and 3 are designated by like reference numerals having added 200 to the reference numerals in the embodiment of figure 3.

25 In the embodiment shown in figure 3 network node 302 comprises portable computer 320 which is physically coupled to the computer network through docking station 322. For example portable computer 320 is a lap top computer, palm top computer or other mobile computing device.

30 In operation portable computer 320 is reconnected to the computer network by inserting portable computer 320 into docking station 322 and booting portable computer 320. Next a log on dialogue is started between network server 300 and portable computer 320.

If a Microsoft windows operating system is used network server 300 plays the role of a domain controller for portable computer 320. Log on request 306 is processed by program component 304 of network server 300. When the user of portable computer 320 has correctly entered a valid user ID and password portable computer 320 is logged on and is thus logically reconnected to the network.

As a consequence program component 304 invokes program component 308 for initiation of the discovery procedure for portable computer 320. The IP address of portable computer 320 which is known to network server 300 is transmitted as part of discovery request 310 to discovery server 312 which performs a discovery procedure analogous to the discovery procedure as explained above with respect to figures 1 and 2.

15

Figure 4 shows a further preferred embodiment where like elements are referenced by like reference numerals as in figure 1 having added 300 to the reference numerals.

20 In the embodiment shown in figure 4 network server 400 is a DHCP server. DHCP provides automated configuration services including automated assignment of IP addresses. Network server 400 has a corresponding DHCP services program component 404.

25 In the preferred embodiment considered here, network node 402 is a computer peripheral device, such as a printer. Computer peripheral 402 has a media access control (MAC) address which is stored in permanent storage 420 of computer peripheral 402.

30 Network server 400 has time stamp program component 422. Time stamp program component 422 serves to provide a time stamp for discovery request 410 which is generated by program component 408 for discovery initiation.

In operation when computer peripheral 402 is physically coupled to the computer network an IP address request 406 is automatically generated by computer peripheral 402 based on the MAC address stored in storage 420 in accordance with the DHCP protocol. In response to IP address request 406 network server 400 invokes DHCP services program component 404 for processing of IP address request 406.

DHCP services program component 404 assigns an IP address to computer peripheral 402. Next program component 408 is invoked which generates discovery request 410 which includes the newly assigned IP address of computer peripheral 402. Discovery request 410 is time stamped by time stamp program component 422 and then sent from network server 400 to discovery server 412.

Discovery server 412 has a discovery log file 424 in which the IP addresses and time stamps of previously received discovery requests for which discovery procedures have been performed are stored. For example discovery log 424 has a tabular structure, e.g. for each request a request number, time stamp and IP address are stored in the discovery log 424. Discovery server 412 queries discovery log 424 using the IP address of discovery request 410 as a key. If an entry in discovery log 424 with that IP address is identified the corresponding time stamp stored in the discovery log 424 and the time stamp of discovery request 410 are compared:

If time stamp of discovery request 410 is a predetermined amount of time later than the time stamp stored in discovery log 424 the IP address of discovery request 410 is entered into memory 414 in order to initiate a corresponding discovery procedure by discovery program 416. If the contrary is the case discovery request 410 is refused. This way unnecessary discovery procedures are avoided when computer peripheral 402 is frequently removed and reconnected to the computer network. For example the predetermined amount of time can be set to 6, 12 or 24 hours.

As an alternative to time stamping discovery request 410 by time stamp program component 422, the reception time of discovery request 410 can be used as a time reference. In this instance time stamp program component 422 is not
5 required. This minimizes the complexity of the DHCP Server 400.

Again, it is to be noted that network server 400 and discovery server 412 are implemented on different server computers; however, network server 400 and discovery server 412 can also be implemented on the same server computer
10 which is partitioned correspondingly.

Figure 5 shows a corresponding flow chart. In step 500 an access request is received by a network server from a network node which has been physically reconnected to the network. In step 502 the access request is processed by the
15 network server. After successful completion of the processing in step 502, i.e. when the network node is logically reconnected to the network, a discovery request is generated in step 504 and is time stamped with a time stamp i.

In step 506 the time stamped discovery request is sent to the discovery server.
20 In response the discovery server performs a discovery procedure and enters the node ID of the network node for which the discovery procedure has been performed together with time stamp i in a discovery log file.

In step 508 the network node is removed from the network. In step 510 the network node is reconnected to the network. As consequence a renewed access request from the reconnected network node is received by the network server in
25 step 512 which is processed in step 514 and time stamped with time stamp j in step 516 before the time stamp discovery request is sent to the discovery server in step 517.

30 When the renewed discovery request is received by the discovery server the discovery server checks if time stamp j minus time stamp i is above a prede-

fined threshold. In other words the discovery server checks if a sufficient amount of time has lapsed between the first discovery request for which a discovery procedure has been performed in step 507 and the subsequent discovery request with time stamp j. This check is performed in step 518.

5

If the amount of time which has lapsed between the two subsequent discovery requests is below the threshold the control goes to step 519 where the renewed discovery request is rejected. If the contrary is the case a renewed discovery procedure is carried out in step 520.

10

list of reference numerals

5	100	network server
	102	network node
	104	program component
	106	access request
	108	program component
10	110	discovery request
	112	discovery server
	114	memory
	116	discovery program
	118	poll
15	300	network server
	302	network node
	304	program component
	306	log on request
	308	program component
20	310	discovery request
	312	discovery server
	314	memory
	316	discovery program
	318	poll
25	320	portable computer
	322	docking station
	400	network server
	402	computer peripheral
	404	DHCP services program component
30	406	IP address request
	408	program component
	410	discovery request

	412	discovery server
	414	memory
	416	discovery program
	418	poll
5	420	storage
	422	time stamp program component
	424	discovery log

c l a i m s

5 1. A method of discovery of a network node, the method comprising the steps of:

- 10 - transmitting an access request from the network node to a network server of a computer network for coupling of the network node to the computer network,
- transmitting a discovery request from the network server to a discovery server, the discovery request comprising an identifier of the network node,
- 15 - performing a discovery procedure of the network node by the discovery server using the identifier.

20 2. The method of claim 1, further comprising processing of the access request by the network server and generating of the discovery request by the network server after acceptance of the access request.

3. The method of claim 1, whereby the access request is a log on request.

25 4. The method of claim 3, the network server being a domain controller.

5. The method of claim 1, whereby the access request is an internet protocol address request.

30 6. The method of claim 5, the network server being a dynamic host configuration protocol server.

7. The method of claim 1, further comprising the steps of:

- storing of a time stamp of the access request by the network server,
- when a subsequent access request is received by the network server:
transmitting a subsequent discovery request in response to the subsequent access request only if the subsequent access request is distanced by at least a predetermined amount of time from the time stamp.

8. The method of claim 1, the discovery procedure comprising polling of the network node in order to discover network topology, network node type, network node status and / or network node configuration information.

9. A method of coupling a user device to a computer network, the method comprising the steps of:

- receiving an access request from the user device by a network server of the computer network for coupling of the user device to the computer network,
- sending a discovery request from the network server to a discovery server in order to request a discovery procedure to be performed by the discovery server for the network node.

10. The method of claim 9, the user device being a portable client computer and the access request being a log on request.

11. The method of claim 9, the user device being a computer peripheral and the access request being an internet protocol address request.

12. The method of claim 9, further comprising:

- receiving a subsequent access request from the network node for renewed coupling of the network node to the computer network,
- determining whether the renewed access request is distanced from the access request by at least a predetermined amount of time; and if this condition is fulfilled,
- sending of a renewed discovery request to the discovery server to request a renewed discovery procedure to be performed for the network node.

13. The method of claim 9, further comprising processing of the access request by the network server and generating the discovery request by the network server if the access request has been accepted by the network server.

14. The method of claim 9, further comprising temporarily coupling the network node to the computer network by means of a docking station.

15. A computer program product for initiating a discovery procedure for a network node, the computer program product comprising program means for performing the steps of:

- receiving an access request from the network node for coupling of the network node to the computer network,
- generating of a discovery request for initiating a discovery procedure for the network node in response to the access request.

16. The computer program product of claim 15, the program means being adapted to determine whether a subsequent access request, which is received after the access request, is distanced from the access request by at least a predetermined amount of time, whereby the discovery request is only generated if this condition is fulfilled.

17. The computer program product of claim 15, the program means being adapted to time stamp the access request.

18. A computer program product for performing a discovery procedure, the computer program product having program means for performing the steps of:

- receiving a discovery request from a network server, the discovery request comprising an identifier of a network node,
- performing a discovery procedure for the network node using the identifier.

19. The computer program product of claim 18, the program means being adapted to store identifiers of network nodes in a buffer memory.

20. The computer program product of claim 19, the program means being adapted to perform the discovery procedure by polling of the network node.

21. A network server for coupling of a network node to a computer network, the network server adapted to:

- receiving an access request from the network node and

sending a discovery request to a discovery server in response to the access request, the discovery request comprising an identifier of the network node.

- 5 22. A domain controller comprising an authentication component for processing of an access request from a network node and a discovery initiation component for generating a discovery request in response to successful authentication of the network node by the authentication component.

- 10 23. The domain controller of claim 22, the access request being a log on request and the authentication component being a log on component.

24. A dynamic host configuration protocol server comprising:

- 15 - a dynamic host configuration protocol component for providing an IP address to a network node in response to an IP address request received from the network node,
- 20 - a discovery initiation component for generating a discovery request in response to the IP address request, the discovery request comprising the IP address in order to enable a discovery server to perform a discovery procedure for the network node identified by the IP address.

- 25 25. The dynamic host configuration protocol server of claim 24, the discovery initiation component being adapted to generate the discovery request only if at least a predetermined amount of time has passed since a previous discovery request for the network node.

26. A discovery server comprising:

- 30 - a memory component for storing an IP address received as part of a discovery request from a dynamic host configuration protocol server,

- a discovery program component for performing a discovery procedure of a network node being identified by the IP address stored in the memory component in response to the discovery request.

5

26. The discovery server of claim 25, the discovery program component being adapted to perform the discovery procedure by polling of the network node.

10

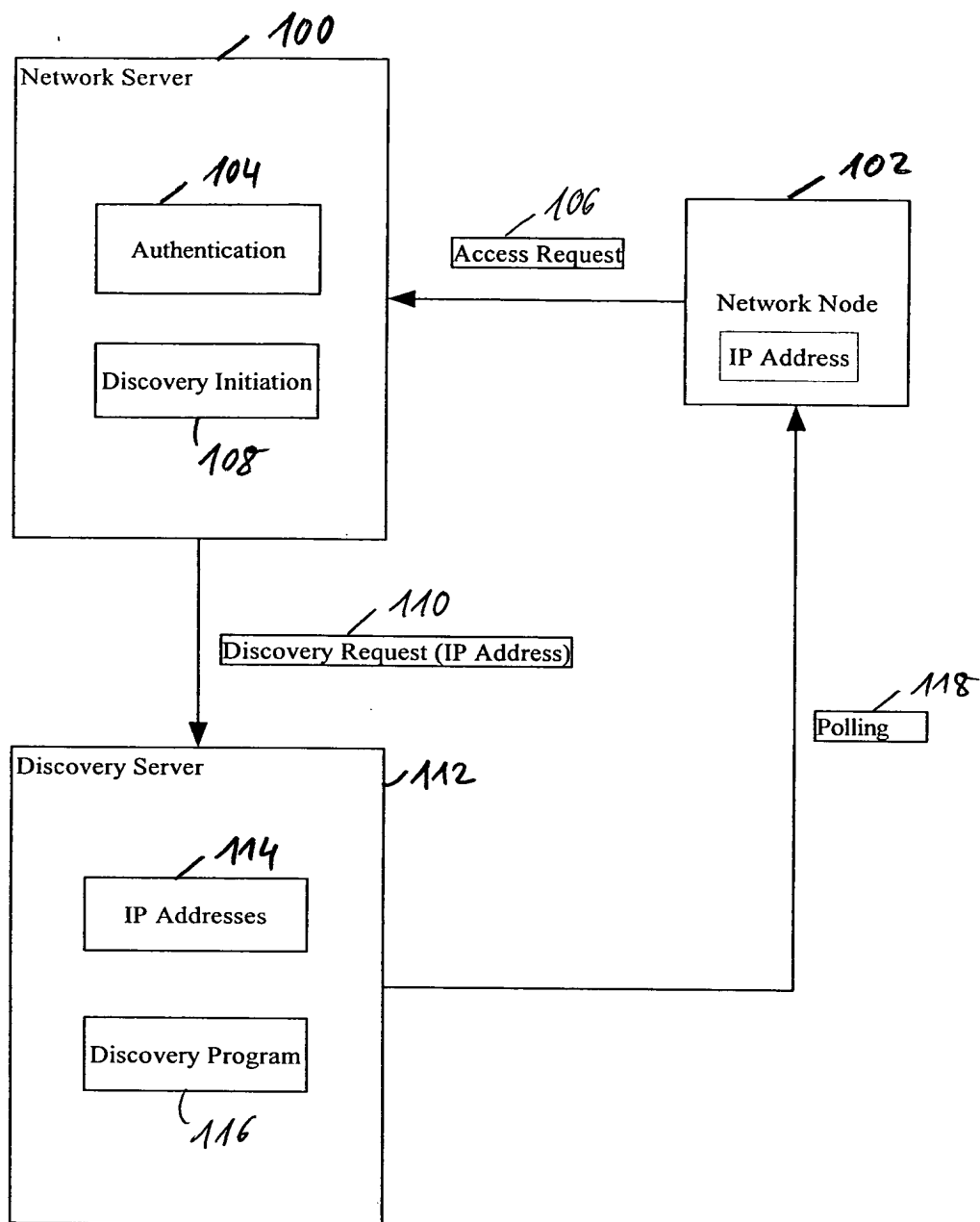


Fig. 1

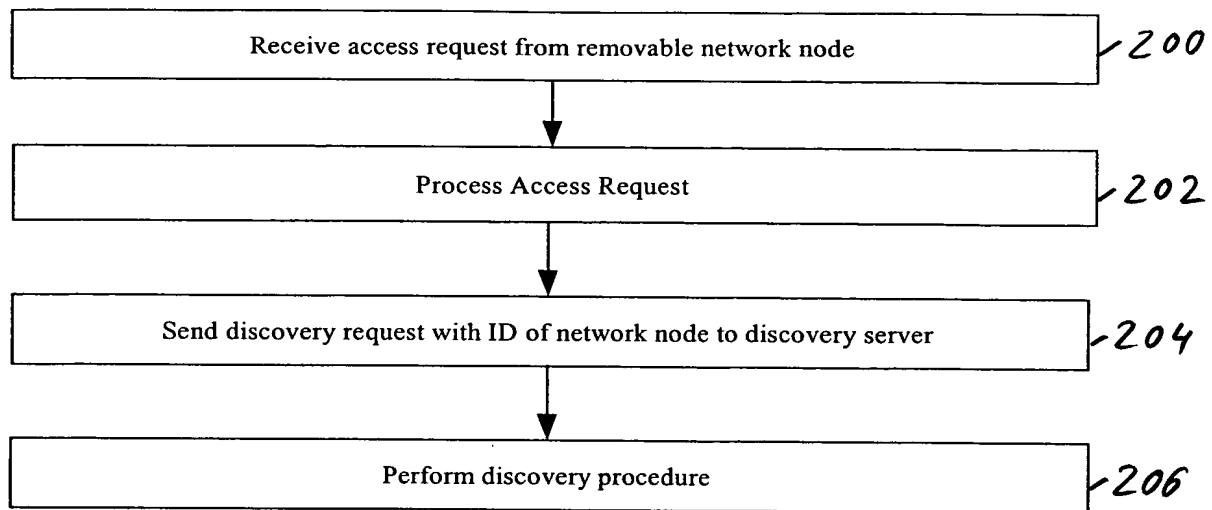


Fig. 2

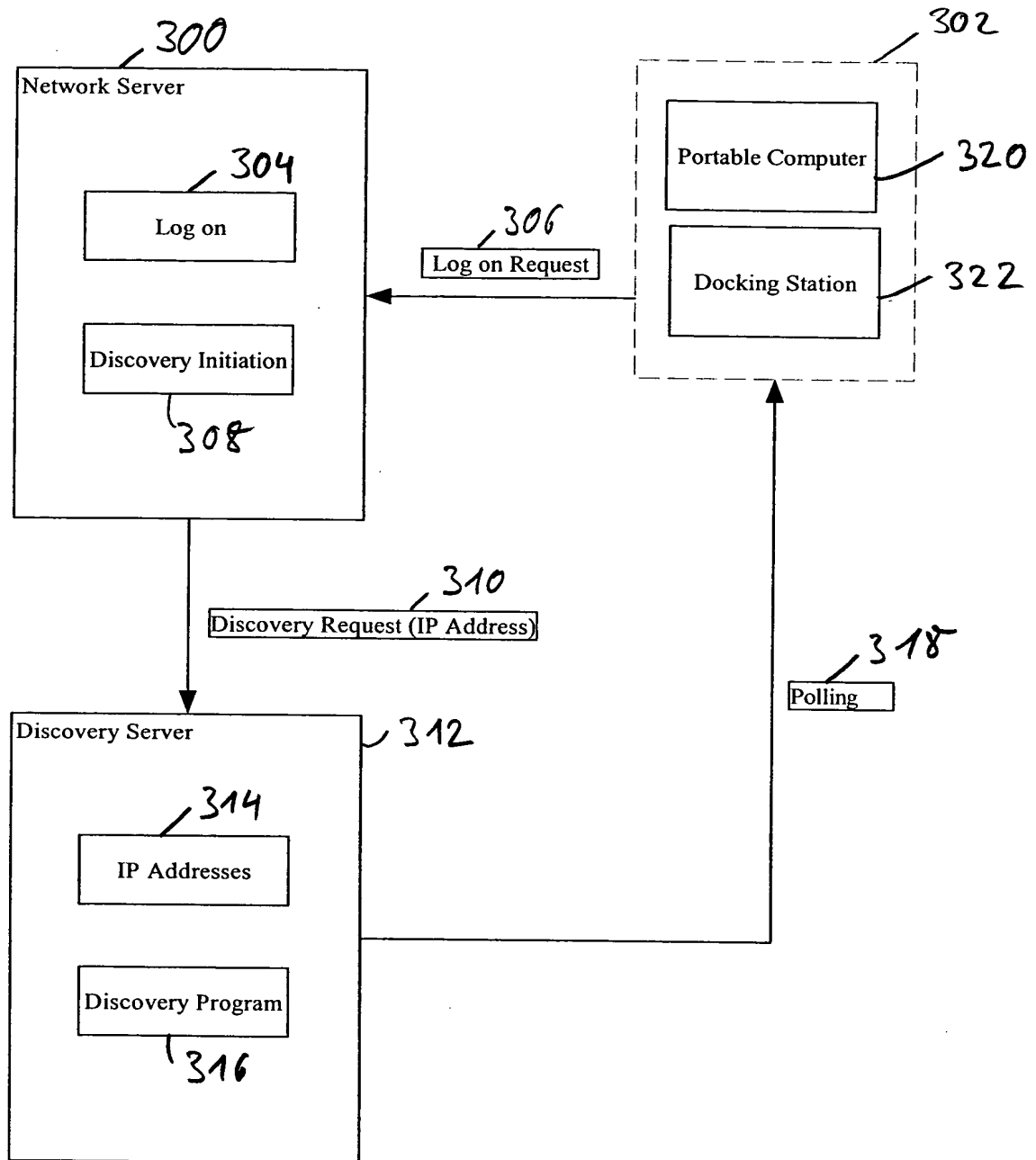


Fig. 3

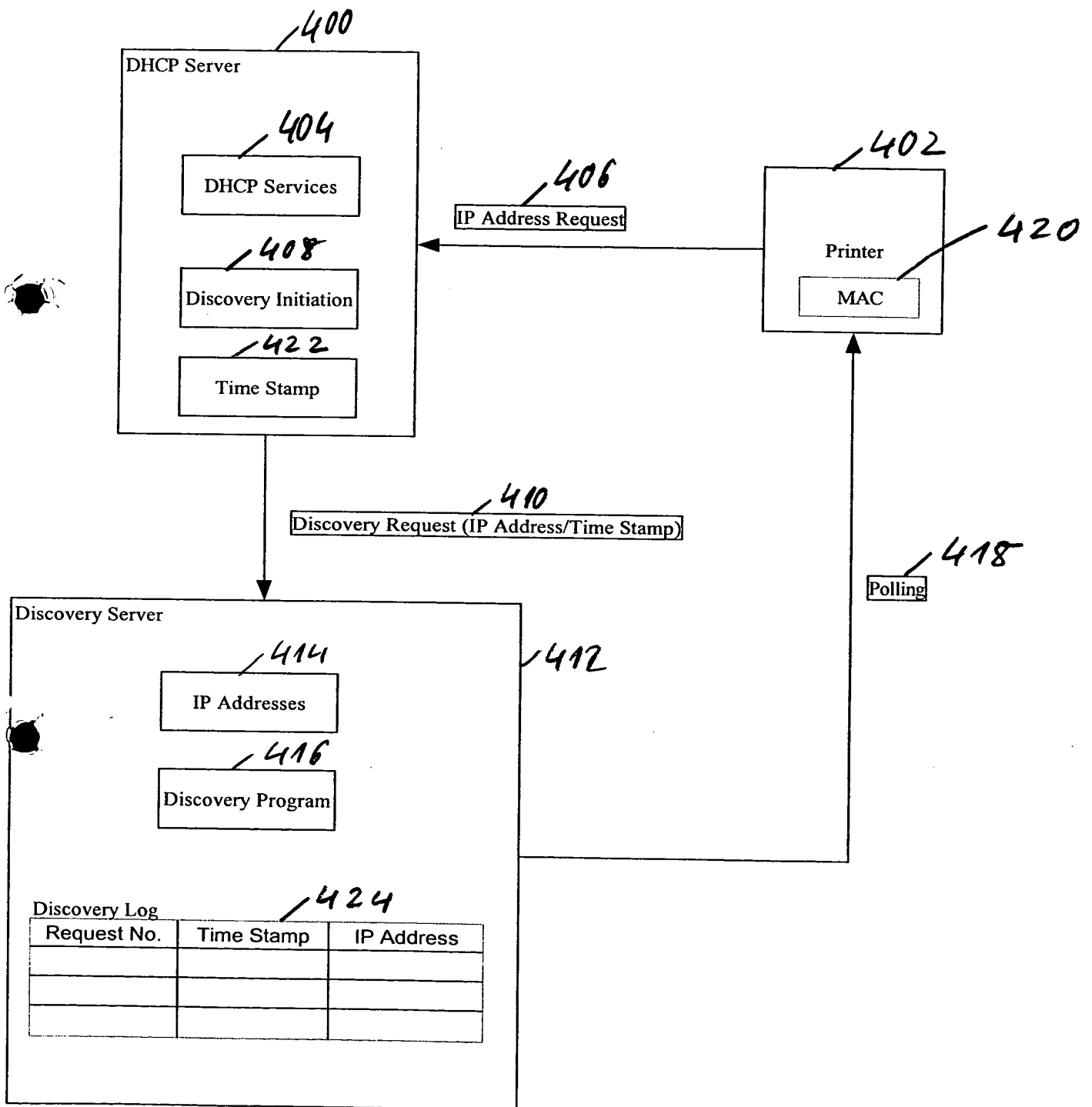


Fig. 4

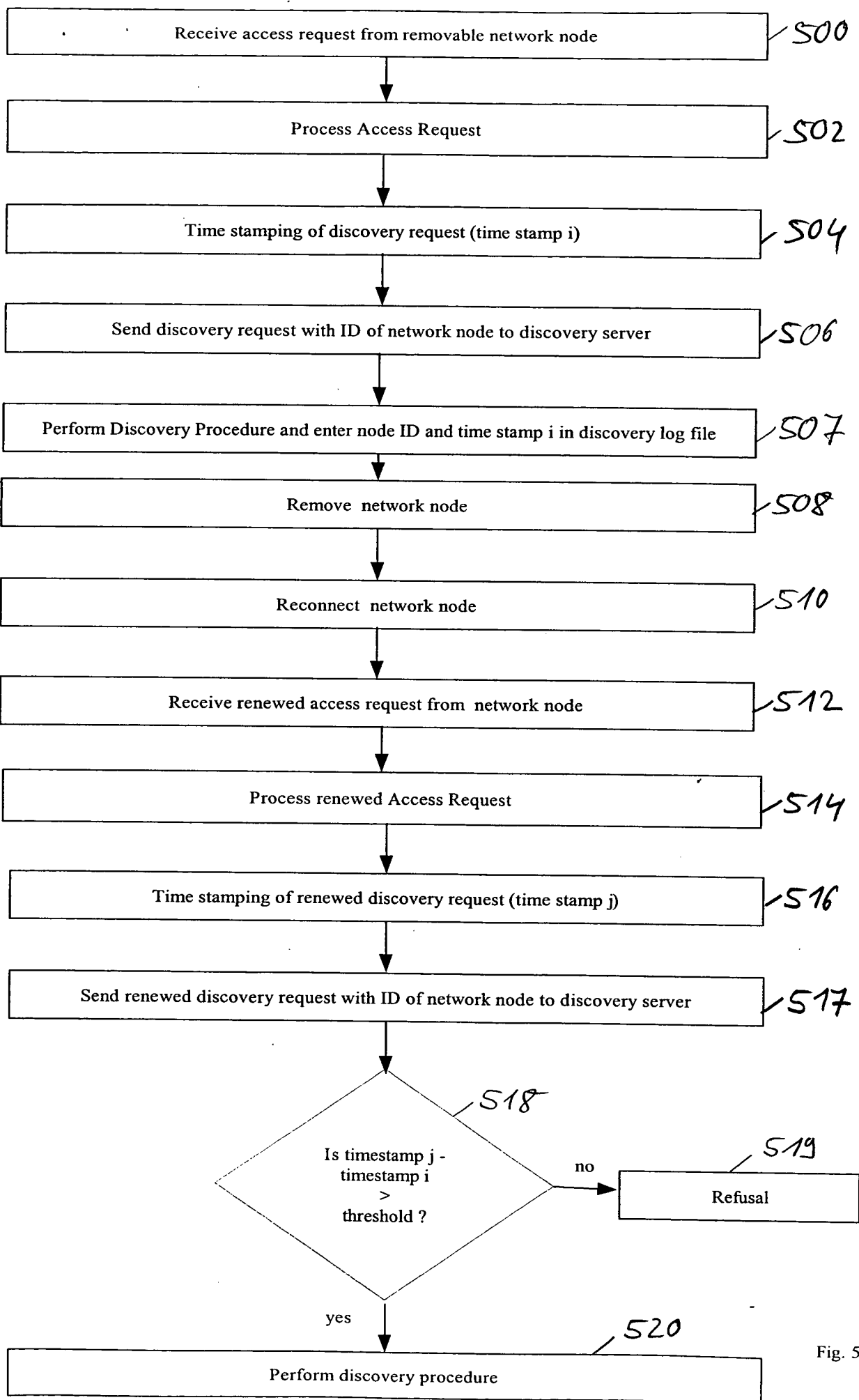


Fig. 5

AbstractNetwork server and method of discovery of a network node

5

The present invention relates to discovery of a network node. In contrast to the prior art discovery is not performed in accordance with a prescheduled discovery scheme but discovery is triggered when a network node is reconnected to the computer network. The discovery procedure is initiated by an access request of the network node, such as a log on request or an IP address request.

10

(Figure 1)

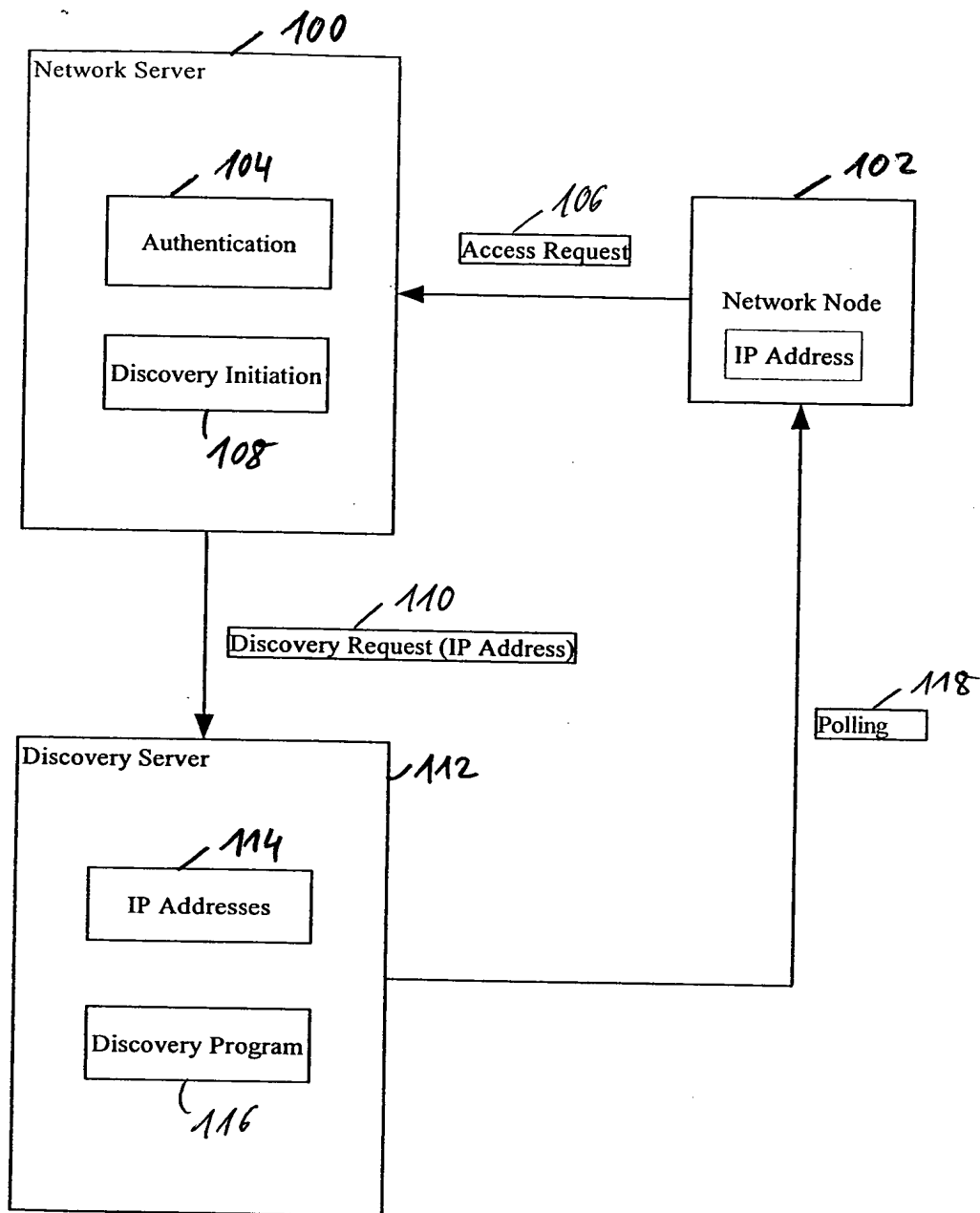


Fig. 1